Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer program product tangibly embodied in machine-readable storage device, for using incremental generation to develop applications, the computer program product being operable to cause data processing apparatus to:

store a plurality of metadata development objects corresponding to a metamodel, wherein the development objects comprise a plurality of development objects specified as main development objects;

identify a first main development object;

identify main development objects related to the first main development object[s] by an aggregation relationship, wherein an aggregation relationship is a whole-part relationship between an aggregate and one or more constituent parts;

identify main development objects related to the first main development object by an association relationship, wherein an association relationship represents a connection between between two classes;

determine if any identified main development objects have changed; and

re-generate the first main development object if any identified main development objects have changed[;],

wherein the main development objects are persisted in separate files and correspond to file borders; and

wherein identifying main development objects related to the first main development objects by the association relationship comprises identifying main development objects related to the first main development objects by the association relationship by following association relationships only across a single file border to a next main development object.

2. (Original) The computer program product of claim 1, wherein identifying main development objects related to the first main development objects by the aggregation relationship comprises identifying main development objects related to the first main development objects by the aggregation relationship by following all aggregation relationships to other main development objects.

3. (Canceled).

4. (Original) The computer program product of claim 1, wherein identifying main development objects related to the first main development objects by the association relationship comprises identifying main development objects related to the first main development objects by the association relationship by following aggregation relationships and then a subsequent association relationship only to a next main development object.

5. (Original) The computer program product of claim 1, further comprising instructions to cause data processing apparatus to:

identify a second main development object,

wherein identifying the first main development object comprises identifying the first main development object that is a root main development object associated with the second main development object.

6. (Previously Presented) The computer program product of claim 5, wherein identifying the second main development object comprises identifying a non-main development object as a starting point and asking the non-main development object what its main development object is.

7. (Original) The computer program product of claim 5, further comprising instructions to cause data processing apparatus to:

for each child main development object of the root main development object,

identify a child main development object that has not been considered;

identify main development objects related to the child main development objects by an

aggregation relationship;

identify main development objects related to the child main development object by an association relationship;

determine if any identified main development objects have changed; and

re-generate the child main development object if any identified development objects have changed.

8. (Canceled)

9. (Canceled)

10. (Currently Amended) The computer program product of claim [8]1, further comprising instructions to cause data processing apparatus to identify all main development objects that have changed based on identified files that have changed.

11. (Currently Amended) The computer program product of claim 10, wherein the identified files that have changed are identified by a resource tree traversal or a notification from [a]the metamodel to which the main development objects correspond.

12. (Original) The computer program product of claim 1, wherein the first main development object corresponds to a development object identified by a development tool making a request for re-generation.

13. (Currently Amended) A system for using incremental generation to develop applications, the system comprising:

a repository for storing a plurality of metadata development objects corresponding to a metamodel, wherein the development objects comprise a plurality of development objects specified as main development objects;

a generator comprising;

a search module configured to identify a first main development object, to identify main development objects related to the first main development objects by an

aggregation relationship, wherein an aggregation relationship is a whole-part relationship between an aggregate and one or more constituent parts, and to identify main development objects related to the first main development object by an association relationship, wherein an association relationship represents a connection between between two classes;

a comparator module configured to determine if any identified main development objects have changed; and

a generation module configured to re-generate the first main development object if any identified development objects have changed[;],

wherein the main development objects are persisted in separate files and correspond to file borders; and

wherein identifying main development objects related to the first main development objects by the association relationship comprises identifying main development objects related to the first main development objects by the association relationship by following association relationships only across a single file border to a next main development object.

14. (Original) The system of claim 13, further comprising a development tool configured to identify files that have changed, wherein the comparator module is further configured to determine if any identified main development objects have changed based on information from the development tool identifying files that have changed.

15. (Original) The system of claim 13, further comprising a development tool, wherein the comparator module is further configured to receive an indication of the first main development object from the development tool.

16. (Original) The system of claim 13, further comprising a cache, wherein the comparator module is further configured to store the identified main development objects in the cache.

17. (Currently Amended) A method for using incremental generation to develop applications, the method comprising:

storing a plurality of metadata development objects corresponding to a metamodel;

identifying a first main development object;

identifying main development objects related to the first main development objects by an aggregation relationship, wherein an aggregation relationship is a whole-part relationship between an aggregate and one or more constituent parts;

identifying main development objects related to the first main development object by an association relationship, wherein an association relationship represents a connection between between two classes;

determining if any identified main development objects have changed; and

re-generating the first main development object if any identified main development objects have changed[;],

wherein the main development objects are persisted in separate files and correspond to file borders; and

wherein identifying main development objects related to the first main development objects by the association relationship comprises identifying main development objects related to the first main development objects by the association relationship by following association relationships only across a single file border to a next main development object.

18. (Original) The method of claim 17, wherein identifying main development objects related to the first main development objects by the aggregation relationship comprises identifying main development objects related to the first main development objects by the aggregation relationship by following all aggregation relationships to other main development objects.

19. (Canceled).

20. (Original) The method of claim 17, further comprising:

identifying a second main development object, wherein identifying the first main development object comprises identifying the first main development object that is a root main development object associated with the second main development object.

21. (Original) The method of claim 20, further comprising:

for each child main development object of the root main development object,

identifying a child main development object that has not been considered;

identifying main development objects related to the child main development objects by an aggregation relationship;

identifying main development objects related to the child main development object by an association relationship;

determining if any identified main development objects have changed; and

re-generating the child main development object if any identified development objects have changed.